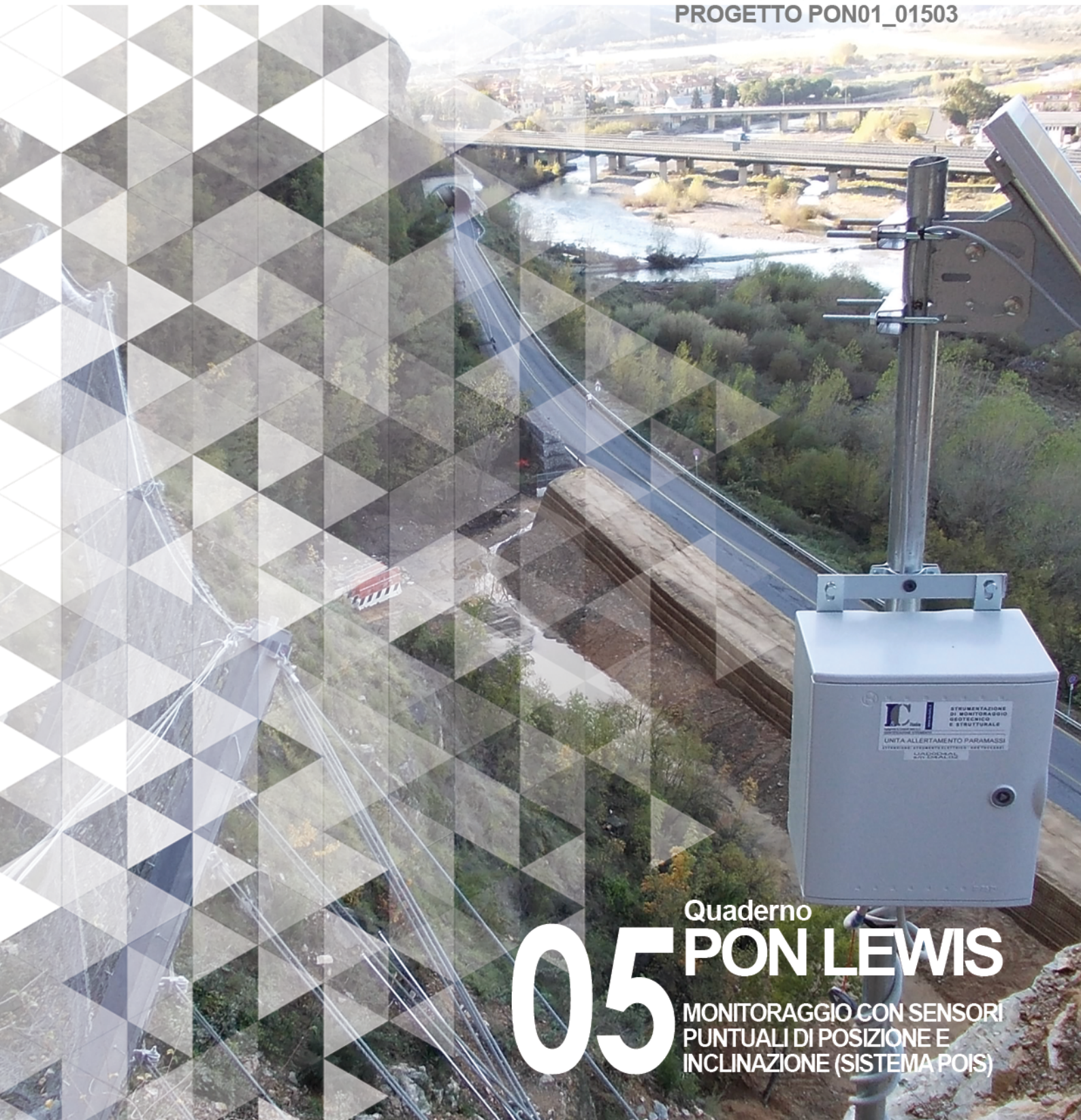


SISTEMI INTEGRATI PER IL MONITORAGGIO, L'EARLY WARNING
E LA MITIGAZIONE DEL RISCHIO IDROGEOLOGICO
LUNGO LE GRANDI VIE DI COMUNICAZIONE



investiamo nel vostro futuro
PROGETTO PON01_01503



05

Quaderno PON LEWIS

MONITORAGGIO CON SENSORI
PUNTUALI DI POSIZIONE E
INCLINAZIONE (SISTEMA POIS)





autostrade//Tech



A cura di Giuseppe Artese | **DELIVERABLE WP 2.4**

**Monitoraggio con sensori puntuali di posizione e inclinazione
(Sistema POIS)**

Sistemi integrati per il monitoraggio, l'early warning e la mitigazione del rischio idrogeologico lungo le grandi vie di comunicazione"

Premessa

Frane e inondazioni sono un problema di grande rilevanza nel nostro Paese. Negli ultimi anni le vittime e i danni dei disastri idrogeologici hanno raggiunto livelli inaccettabili e impongono un grande e immediato impegno della comunità nazionale per cercare di mitigare il livello di rischio, utilizzando strategie articolate ed efficaci capaci di integrare, in una visione organica, interventi strutturali e non strutturali.

Su questi temi l'Università della Calabria è impegnata da anni in attività di studio e di ricerca di rilevanza nazionale e internazionale e nella diffusione e promozione della cultura della previsione e prevenzione del rischio idrogeologico. Nel 2011 insieme ad altri partner, ha promosso un progetto di ricerca triennale, "Sistemi integrati per il monitoraggio, l'early warning e la mitigazione del rischio idrogeologico lungo le grandi vie di comunicazione", finalizzato allo sviluppo di un sistema complesso e articolato di preannuncio delle frane da impiegare per le fasi di previsione/prevenzione del rischio idrogeologico.

Il Progetto, indicato con l'acronimo LEWIS (Landslide Early Warning Integrated System), è stato svolto, nel periodo 2012-2014, nel quadro del Programma Operativo Nazionale 2007-13 "Ricerca e Competitività".

I risultati conseguiti sono descritti in questa collana di Quaderni PON LEWIS.

Il progetto è stato sviluppato dall'Università della Calabria e Autostrade Tech S.p.A. insieme ai partner industriali Strago e TDGroup, alle Università di Catania, di Reggio Calabria e di Firenze e al CINID (Consorzio Interuniversitario per l'Idrologia). Per l'Ateneo calabrese hanno partecipato diversi laboratori e gruppi di ricerca: CAMILab (con funzione di coordinamento), μ Wave, Geomatica, Nems, Geotecnica, Dipartimento di matematica.

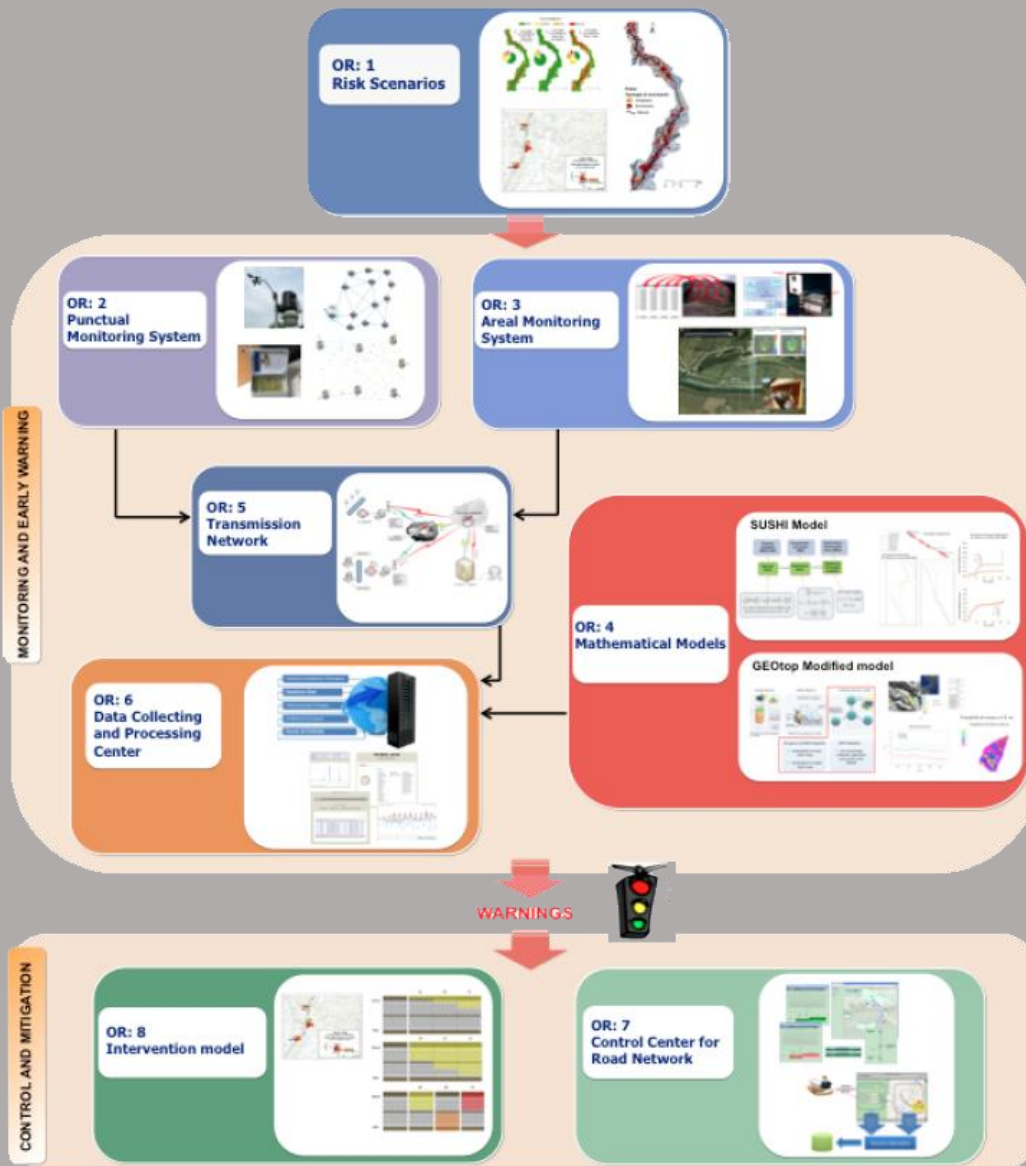


Figura 1 - Articolazione del sistema integrato di monitoraggio dei versanti e di preannuncio dei movimenti franosi

Il progetto è finalizzato allo sviluppo di un sistema di monitoraggio dei versanti e di preannuncio dei movimenti franosi che possono interessare le grandi vie di comunicazione e all'identificazione dei conseguenti interventi non strutturali di mitigazione.

Il sistema è articolato in due sottosistemi (fig. 1):

- ✓ Monitoraggio e preannuncio,
- ✓ Controllo e mitigazione,

che richiedono la preventiva individuazione degli scenari di rischio ossia dei danni che l'eventuale attivazione di una frana può produrre sugli elementi a rischio presenti (infrastruttura viaria, autoveicoli, persone). La procedura originale sviluppata nell'ambito del progetto prevede l'identificazione, lungo il tratto autostradale di interesse, delle aree soggette a movimenti franosi e la conseguente definizione dei relativi scenari di evento e di rischio.

Il sottosistema *Monitoraggio e preannuncio* è formato da diverse componenti: rete di monitoraggio "puntuale" che comprende sensori che misurano localmente l'inizio degli spostamenti superficiali o profondi; rete di monitoraggio "areale" che include sensori che controllano a distanza il fenomeno franoso con tecniche radar; modelli matematici di simulazione dell'innesco e della propagazione dei movimenti franosi. Nel progetto LEWIS sono state sviluppate numerose componenti innovative e sono state modificate e migliorate altre componenti già esistenti. In particolare tra i sensori puntuali sono stati sviluppati i sistemi SMAMID e POIS; tra quelli areali sono stati realizzati un radar in banda L, uno scatterometro, un interferometro; tra i modelli si sono sviluppati e/o migliorati: GEOtop, SUSHI, SCIDDICA.

La raccolta dei dati misurati dai sensori è affidata ad un unico sistema di trasmissione dati che trasmette anche le informazioni necessarie per il funzionamento dei modelli. Il sottosistema è completato da un Centro di acquisizione ed elaborazione dei dati (CAED) che, sulla base dei dati misurati dai sensori e delle indicazioni dei modelli, valuta la situazione di pericolo lungo il tronco autostradale emettendo i relativi livelli di criticità.

I livelli di criticità emessi dal CAED sono l'elemento di collegamento tra il sottosistema *Monitoraggio e preannuncio* e il sottosistema *Controllo e mitigazione*. Gli avvisi di criticità sono acquisiti dal Centro di comando e controllo del traffico (CCCT) che, sulla base di un modello di intervento predefinito, attiva le procedure standardizzate per la mitigazione del rischio, che vanno dalla

sorveglianza diretta del tratto di interesse da parte di squadre tecniche all'interruzione del traffico su entrambe le direzioni di marcia.

Il progetto prevede anche lo sviluppo di attività sperimentali su tre tronchi autostradali lungo la A3, la A16 e la A18, nonché l'erogazione di un Master di secondo livello denominato ESPRI (ESperto in Previsione/Prevenzione Rischio Idrogeologico).

Il progetto di ricerca è stato organizzato in Obiettivi Realizzativi (OR), ciascuno dei quali suddiviso in Work Package (WP), a loro volta articolati in Attività Elementari (AE). In totale erano previste 11 OR, 47 WP e 243 AE. In particolare le OR 1-8 riguardano la ricerca e si articolano in 26 WP e 139 AE. Le OR 9-11 sono dedicate a sperimentazione, governance e trasferimento tecnologico, integrazione e aggiornamento dell'attività di ricerca nella fase di Sviluppo Sperimentale e si articolano complessivamente in 21 WP e 104 AE.

I Quaderni che compongono questa collana sono stati costruiti con riferimento ai singoli WP, per la parte che riguarda la ricerca, e quindi ogni Quaderno contiene la descrizione dei risultati conseguiti nel WP, articolata in base alle AE previste.

Sono, inoltre, previsti altri tre Quaderni:

Quaderno 0 che contiene una descrizione di sintesi, in inglese, dei risultati conseguiti nell'ambito del progetto.

Quaderno 28 che contiene l'informazione relativa alle attività di divulgazione dei risultati scientifici.

Quaderno 29 che contiene la descrizione dei risultati conseguiti con l'attività formativa.

Il Quadro editoriale complessivo è riportato in tabella 1:

QUADERNO	OR	WP	TITOLO
0	-	-	Research outcomes
01 Parte prima	1	1.1	Linee guida per l'identificazione di scenari di rischio
01 Parte seconda	1	1.1	Linee guida per l'identificazione di scenari di rischio
02	2	2.1	Monitoraggio idrogeologico
03 Parte prima	2	2.2	Monitoraggio con unità accelerometriche (Sistema SMAMID)
03 Parte seconda	2	2.2	Monitoraggio con unità accelerometriche (Sistema SMAMID)
04	2	2.3	Circuiti integrati a bassa potenza per sistemi di monitoraggio con unità accelerometriche
05	2	2.4	Monitoraggio con sensori puntuali di posizione e inclinazione (Sistema POIS)
06	3	3.1	Sviluppo di uno scatterometro a risoluzione variabile
07	3	3.2	Elettronica di bordo dello scatterometro ed inclinazione
08	3	3.3	Sviluppo di un radar in banda L
09	3	3.4	Tecniche di analisi e sintesi di segnali radar per la simulazione accurata di scenari complessi
10	3	3.5	Elettronica di bordo del radar in banda L

QUADERNO	OR	WP	TITOLO
11	3	3.6	Sistemi interferometrici radar ad apertura sintetica basati a terra
12	4	4.1	Modello areale per il preannuncio delle frane da innesco pluviale (Modello GEOtop)
13	4	4.2	Modelli completi di versante di tipo puntuale per il preannuncio di movimenti franosi (Modello SUSHI)
14	4	4.3	Modelli di propagazione delle frane tipo colate (Modello SCIDDICA)
15	5	5.1	Rete Wireless di Telecomunicazioni: sviluppo e scelta dei parametri di progetto
16	6	6.1	CAED. Acquisizione dati: architettura del sistema
17	6	6.2	CAED. Elaborazione dei dati
18	7	7.1	CCCT. Progettazione
19	7	7.2	CCCT. Interfaccia verso il centro di acquisizione ed elaborazione dati
20	7	7.3	CCCT. Interfaccia con altre centrali operative e canali di diffusione delle notizie
21	7	7.4	CCCT. Modulo per la presentazione e convalida delle allerte
22	7	7.5	CCCT. Modulo per la gestione delle informazioni di traffico
23	7	7.6	CCCT. Integrazioni con moduli speciali

QUADERNO	OR	WP	TITOLO
24	8	8.1	Definizione del modello di intervento e predisposizione del Piano di Emergenza
25	8	8.2	CCCT. Gestione delivery allerte e attivazione squadre d'intervento
26	8	8.3	CCCT. Gestione percorsi alternativi
27	9	9.1 - 9.11	Sperimentazione
28	10	10.1 - 10.2	Piano di comunicazione e diffusione
29	-	-	Master ESPRI (Esperto in Previsione/Prevenzione Rischio Idrogeologico)

Tabella 1 - Quadro editoriale complessivo della collana di Quaderni PON LEWIS

31 dicembre 2014

Il Responsabile Scientifico del progetto PON LEWIS

Pasquale Versace

INDICE

1 Introduzione

ATTIVITA' ELEMENTARE 2.4.1

1 Deliverable Hardware

2 **2.4.1.1 Elementi che compongono il sensore**

2 **2.4.1.2 Sistema di lettura**

4 **2.4.1.3 Sistema di posizionamento satellitare**

5 **2.4.1.4 Sistema di elaborazione dati**

7 **2.4.1.5 Sistema di trasmissione dati**

9 **2.4.1.6 Telaio di supporto**

12 **2.4.1.7 Fasi di lavorazione**

ATTIVITA' ELEMENTARE 2.4.2

19 Deliverable Software

19 **2.4.2.1 Calibrazione metrica delle fotocamere**

22 **2.4.2.2 Elaborazione delle immagini riprese dalle fotocamere e lettura clinometrica**

26 **2.4.2.3 Elaborazione dei rilievi GNSS**

Introduzione

L'attività relativa al WP2.4 ha riguardato la messa a punto di un sensore puntuali di posizione e inclinazione.

Sono stati, pertanto, ottenuti prodotti hardware e software che vengono di seguito descritti e raggruppati in Deliverable Hardware e Deliverable Software.

ATTIVITÀ ELEMENTARE 2.4.1

Deliverable Hardware

Obiettivo del WP 2.4 era la realizzazione di un sensore puntuale integrato, per la misura ed il monitoraggio di posizione ed inclinazione, caratterizzato da dimensioni contenute, basso peso, basso consumo e basso costo.

Per quanto riguarda il posizionamento, la misurazione è ottenuta per mezzo di un ricevitore GNSS. Le variazioni delle posizioni mutue dei punti monitorati sarà ottenuta con l'approssimazione di alcuni centimetri.

La lettura dell'inclinazione avviene grazie alla lettura ottica di due livelle toriche, usando una sensore CCD digitale ed un software di elaborazione per la valutazione della posizione della bolla, quindi, della determinazione della relativa inclinazione. Il sensore è realizzato su una struttura di supporto, illustrato nel seguito del documento, che può essere fissata sia ad un manufatto da monitorare, sia su palo di sostegno da conficcare nel terreno.

Le fasi della ricerca possono essere così elencate:

- Ricerca di componenti reperibili sul mercato (ricevitori GPS/GLONASS/GALILEO, antenne, minicamera, livelle, batterie);
- Taratura delle livelle e studio della loro sensibilità agli agenti atmosferici ed alla temperatura;
- Realizzazione del supporto basetta con viti calanti, aggancio per camera e dispositivo led, aggancio per GNSS, viti di regolazione, dispositivi di connessione a parete, in orizzontale e su palo;
- Implementazione e montaggio degli elementi hardware sul supporto.

2.4.1.1 ELEMENTI CHE COMPONGONO IL SENSORE.

La progettazione del prototipo è stata orientata alla realizzazione di un sistema di monitoraggio fisso completamente automatico, in grado di inviare le misure rilevate periodicamente o su richiesta di un Centro di Controllo attraverso un protocollo di trasmissione bidirezionale. Il sistema è fornito di un computer, per cui è possibile programmarlo in modo da eseguire autonomamente l'elaborazione dei dati rilevati e di trasmettere, di conseguenza, segnali di allerta laddove si dovessero superare le soglie determinate dal gestore del monitoraggio.

La trasmissione bidirezionale consente anche di variare i parametri impostati inizialmente (orario di acquisizione, durata delle acquisizioni satellitari, soglie per i dati osservati).

Lo strumento è costituito principalmente da quattro sottosistemi:

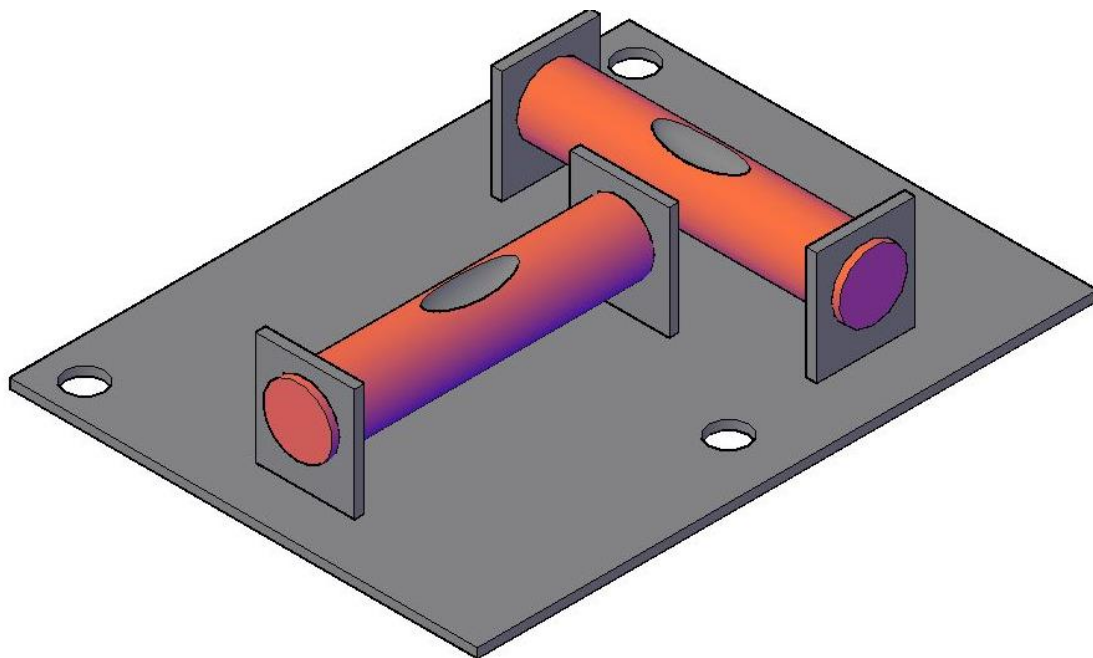
- Sistema di lettura;
- Sistema di posizionamento satellitare;
- Sistema di elaborazione dati;
- Sistema di trasmissione dati;
- Telaio di supporto.

2.4.1.2 SISTEMA DI LETTURA

Il sistema di lettura è di tipo optoelettronico ed è formato da due elementi:

Due Livelle toriche;

Fotocamera.



Le livelle toriche sono disposte a T (vedi figura), con gli assi mutuamente ortogonali.

Sono state scelte livelle di qualità, con fiale in vetro. Per raggiungere l'accuratezza prefissata di $0,001^\circ$, abbiamo adoperato livelle toriche con una sensibilità di $60''/2\text{mm}$. Scegliendo livelle con caratteristiche diverse si può optare per accuratezze maggiori e *range* di acquisizione minori o viceversa.

Per ottenere le inclinazioni, vengono acquisite delle immagini tramite una fotocamera (Raspberry Pi Camera Module Rev 1.3) collegata al sistema di elaborazione dati (Raspberry Pi B 512 MB).

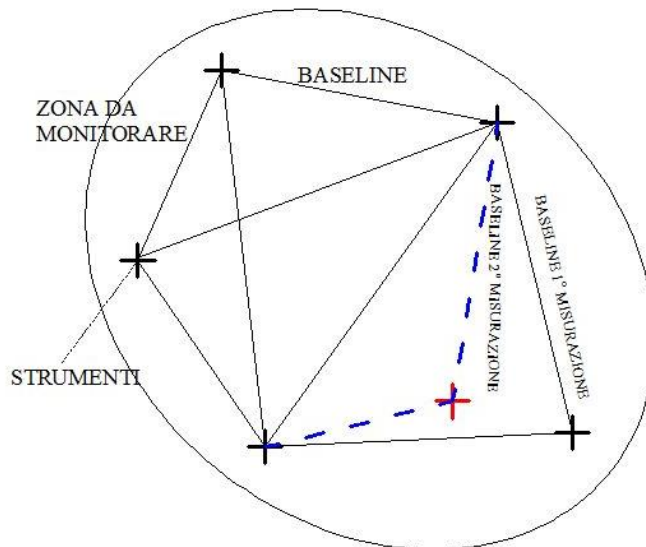


Le caratteristiche della camera sono riportate nella seguente tabella

Risoluzione	5 megapixel (2592 x 1944)
Dimensioni max frame	5 megapixel (2592 x 1944)
Video	1080p30 (Full HD), 720p60 e 640x480p60/90

2.4.1.3 SISTEMA DI POSIZIONAMENTO SATELLITARE

Il posizionamento satellitare viene fatto tramite un ricevitore GNSS. Sono stati adoperati due modelli della ditta Ublox: Ublox NEO-6M e Ublox NEO-6P, quest'ultimo avente l'opzione *ppp* (precise point positioning). Ambedue i ricevitori ricevono la correzione EGNOS, per cui il posizionamento assoluto, con normale visibilità dei satelliti, è di ordine metrico per la planimetria e di un paio di metri per l'altimetria. Come specificato nei report semestrali, lo scopo delle acquisizioni satellitari non è di mero posizionamento, ma ha l'obiettivo di monitorare eventuali variazioni morfologiche della superficie monitorata, controllando variazioni nelle distanze mutue (*baselines*) tra i vari sensori. Nella figura seguente si osserva una variazione delle distanze tra tre vertici, segno di deformazioni nella superficie monitorata.



I ricevitori della serie NEO - 6 M e 6P della Ublox, sono caratterizzati da un ottimo rapporto prezzo-prestazioni e da numerose opzioni di connettività. U-blox 6 M vanta una Time-To-First-Fix (TTFF) di meno di 1 secondo, il che consente di ridurre notevolmente il tempo di attesa per ottenere misure stabili; tale caratteristica è importante per limitare i consumi del nostro sensore. Altra caratteristica positiva di questi ricevitori è data dalla limitazione degli effetti del *multipath*. I ricevitori forniscono anche altre informazioni quali effemeridi, almanacco, ora e stato del satellite e rapporto S/N.

2.4.1.4 SISTEMA DI ELABORAZIONE DATI

Il sistema di elaborazione dati è costituito da una scheda Raspberry Pi B 512 MB. Si tratta di un mini computer multimediale, composto da un processore ARM1176JZ-F a 700 MHz e da un co-processore multimediale VideoCore IV in grado di gestire anche flussi video full HD, abbinato a un quantitativo di memoria RAM pari a 512 MB e a un lettore di schede multiformato oltre ad una uscita HDMI, ad una porta Ethernet e ad una USB, il tutto nelle dimensioni ridotte) ed organizzati in archivi. Le principali caratteristiche sono riportate nella tabella successiva.

Raspberry Pi	Model A	Model B
SoC: ^[1]	<u>Broadcom</u> BCM2835 (CPU + GPU + DSP + SDRAM) ^[2]	
CPU:	700 MHz ARM1176JZF-S core (famiglia ARM11)[2]	
GPU:	Broadcom VideoCore IV,[37] OpenGL ES 2.0, 1080p30 H.264 high-profile decode[2]	
Memory (SDRAM):	256 Megabytes (condivisa con la GPU)	256 o 512 Megabytes (condivisa con la GPU)
USB 2.0 ports:[8]	1	2 (attraverso un hub USB integrato)[35]
Output video:[1]	Connettore RCA per il video composito, HDMI	
Output audio:[1]	3,5 mm jack, HDMI	
Memoria:[8]	SD / MMC / SDIO card slot	
Collegamenti di rete:[1][8]	Nessuno	Ethernet 10/100 (RJ-45)[35]
Periferiche di basso livello:	2x13 header pins for GPIO, SPI, I ² C, UART, +3,3 Volt, +5 Volt[37][38]	
Real-time clock:[1]	No clock or battery	
Corrente (potenza) assorbita[39]:	300 mA, (1,5 W)	700 mA, (3,5 W)
Alimentazione:[1]	5 V via MicroUSB o GPIO header	
Dimensioni:	85,60 mm × 53,98 mm[40] (3.370 inch × 2.125 inch)	
Sistemi operativi supportati:	Debian GNU/Linux, Fedora, Arch Linux[3] e Gentoo[41]	
Sistemi operativi non supportati:	RISC OS[9] (shared source)	

2.4.1.5 SISTEMA DI TRASMISSIONE DATI

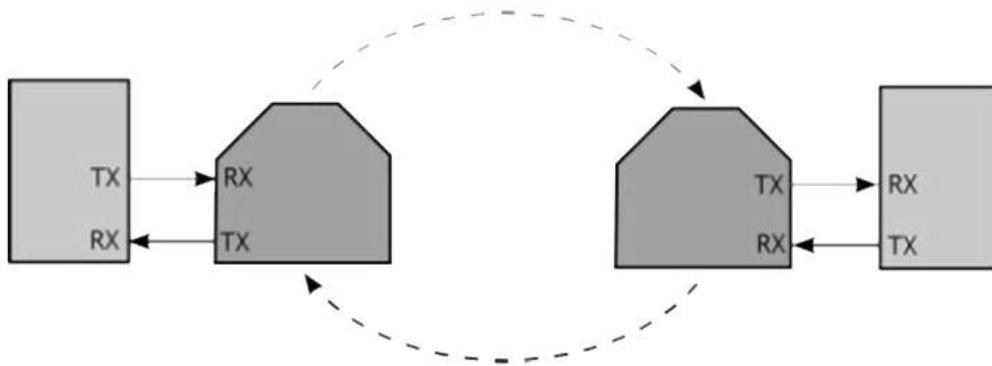
Per la trasmissione dati si è fatto uso di moduli XBee. Essa è una tecnologia radio nata dalla necessità di definire uno standard di comunicazione wireless per la creazione di reti di micro-dispositivi che abbiano bassi costi e consumi limitati.

La rete di sensori utilizza il protocollo ZigBee ed è caratterizzata da:

- Insieme di sensori distribuiti;
- Rete di interconnessione;
- Punto di raccolta dei dati.

Il punto di raccolta dei dati è costituito da un nodo coordinatore. La scheda Raspberry costituisce l'insieme di risorse computazionali al fine di effettuare le correlazioni e l'elaborazione dei dati, il monitoraggio dello stato, ecc.

I moduli XBee sono tra i più utilizzati per le comunicazioni wireless con e tra dispositivi elettronici. Il costo accessibile e le numerose caratteristiche messe a disposizione, tra le quali la possibilità di creare reti, ne fanno prodotti davvero versatili (in grado di applicarsi in diversi campi). Gli XBee sono essenzialmente dei moduli che consentono il trasferimento dati via RF (Radio Frequenza) sfruttando una normale comunicazione seriale. Questo vuol dire che per riuscire a trasferire dei dati, senza fili, è necessario soltanto saper utilizzare la periferica UART inclusa sul nostro microcontrollore o la porta seriale del PC: è possibile difatti immaginare, nella loro configurazione più semplice, una coppia di XBee come un sostituto di un cavo seriale. In breve: tutto ciò che il microcontrollore (o il computer o altro dispositivo in grado di effettuare una comunicazione seriale asincrona) invia sul pin RX di un modulo XBee, giunge ad un altro XBee sul pin TX e può quindi essere ricevuto dal dispositivo remoto.



Il modulo XBee è bidirezionale: un singolo modulo è in grado sia di ricevere che di trasmettere. Un modulo XBee può essere indirizzato in maniera univoca. Ogni XBee ha difatti un numero seriale che lo distingue da tutti gli altri XBee. Tramite una semplice utility è difatti possibile fare in modo che due moduli comunichino soltanto tra di loro senza fare salti mortali nei nostri software. Questo consente di avere molti moduli XBee in una stanza senza che interferiscano tra loro.

Nei moduli XBee sono già implementati nell'hardware tutti i controlli tipici di una comunicazione wireless, come ad esempio i controlli sugli errori di trasmissione, che garantiscono una comunicazione affidabile. Questo, sui moduli economici, siamo costretti a realizzarlo noi via software. Il modulo XBee esegue tutti questi controlli in maniera automatica senza l'intervento dell'utente e possiamo fare in modo che ritrasmetta i dati in automatico se questi non sono giunti a destinazione in maniera corretta.

Sugli XBee è anche possibile assegnare diversi canali ai moduli: avendo più XBee in un'area si può assegnare ad alcuni moduli un canale e ad altri un canale differente: questo minimizza ancor più eventuali interferenze.

I vari modelli di XBee si distinguono tra loro anche in base al tipo di antenna. Antenne più performanti consentono di avere una maggiore potenza di trasmissione e di conseguenza una maggiore distanza coperta. I costi (e i consumi) ovviamente aumentano man mano che il tipo di antenna consente una copertura maggiore. I modelli con minore potenza di trasmissione, ma più economici e compatti, hanno un'antenna stampata su PCB o un'antenna a Chip. Per il nostro sistema abbiamo optato per un modello con antenna a filo, che dovrebbe consentire una portata in campo aperto di oltre un chilometro.



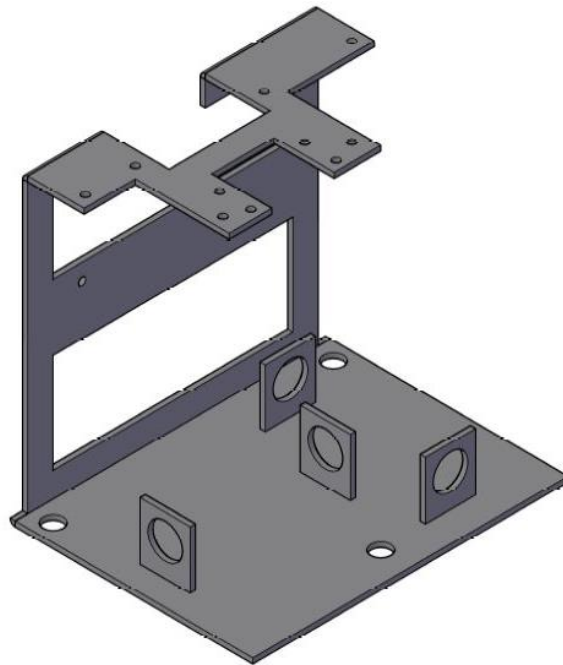
antenna a Chip



antenna a filo

2.4.1.6 TELAIO DI SUPPORTO

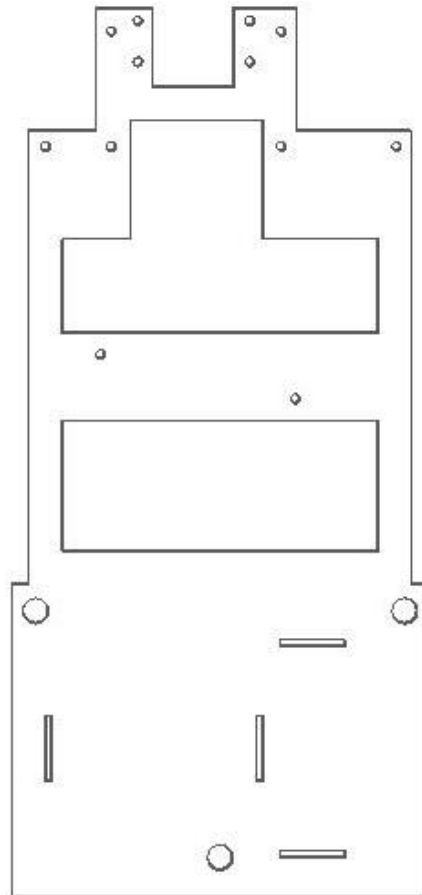
La particolarità del telaio è quella di essere stato realizzato utilizzando un unico pezzo modellato tramite l'apertura di finestre che lo rendono più leggero senza pregiudicarne la rigidità. Sono previsti gli alloggiamenti di tutti gli elementi da inserire (Ricevitore GNSS, XBee, Raspberry, fotocamera, alimentazione, livelle etc.).



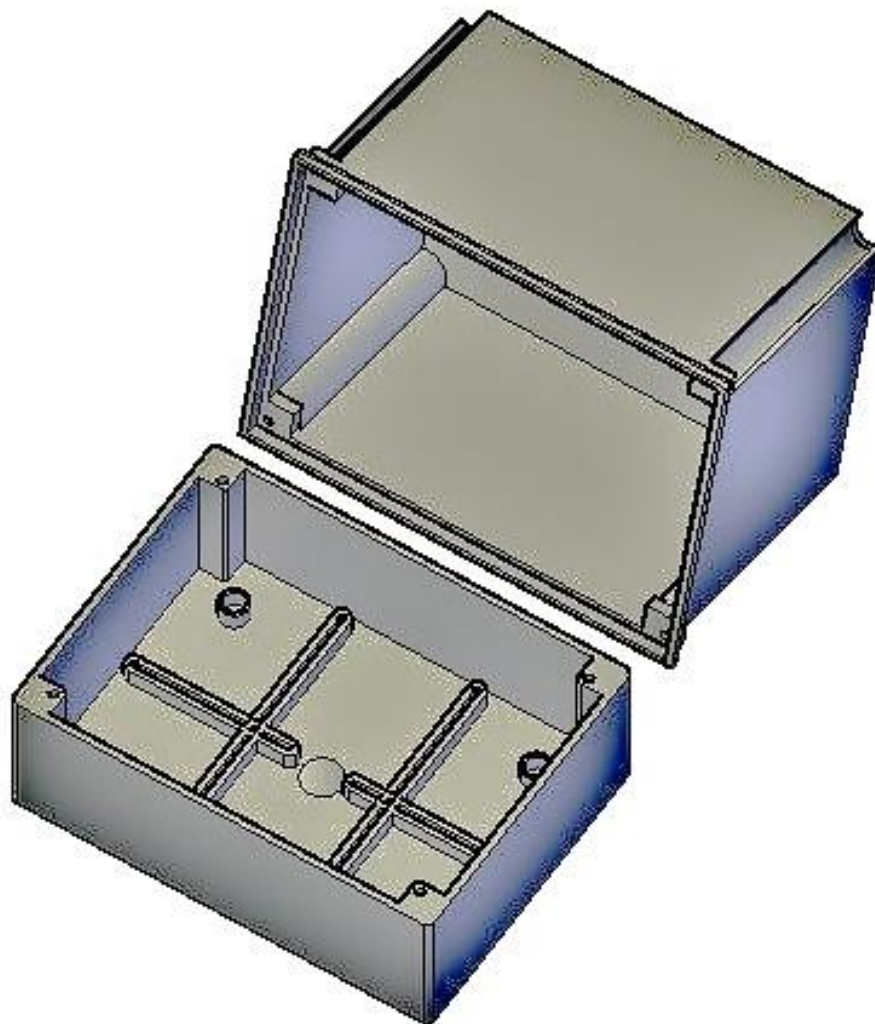
I requisiti richiesti per lo strumento sono:

- Robustezza;
- Leggerezza;
- Semplicità di realizzazione;
- Facilità di messa in opera.

La robustezza è stata garantita dalla scelta di una piastra in acciaio di spessore di 2 mm che gli garantisce un'adeguata rigidità. Alla piastra sono saldati tre prigionieri con viti filettate per il corretto posizionamento iniziale.



Tutto il telaio, infine, è inserito in una cassetta in PVC che garantisce la semplice messa in opera del sensore nella zona da monitorare e la protezione dagli agenti atmosferici (IP 65). Può essere rivestita all'interno con un materiale isolante, dipendentemente dal luogo in cui dev'essere installato.

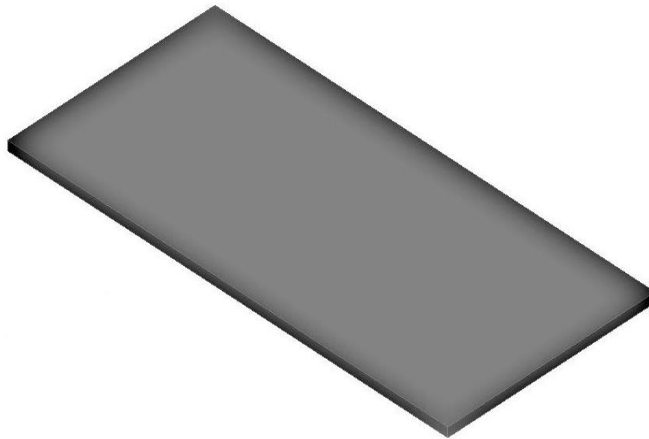


2.4.1.7 FASI DI LAVORAZIONE

1° Fase: Realizzazione piastra base

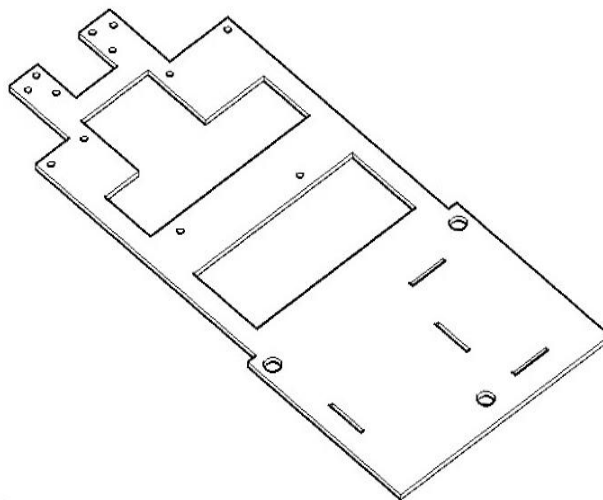
E' utilizzata una piastra in acciaio appositamente dimensionata per garantire una buona rigidezza.

Piastra	
Altezza	268.00 mm
Larghezza	126.00 mm
Spessore	2.00 mm

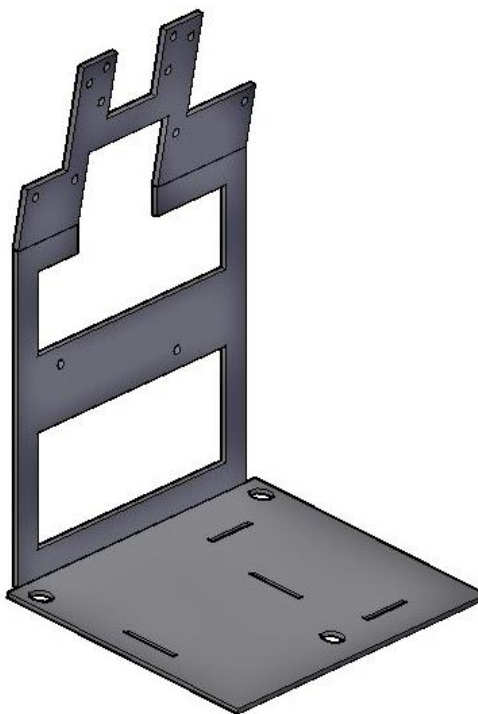


2° Fase: apertura finestre

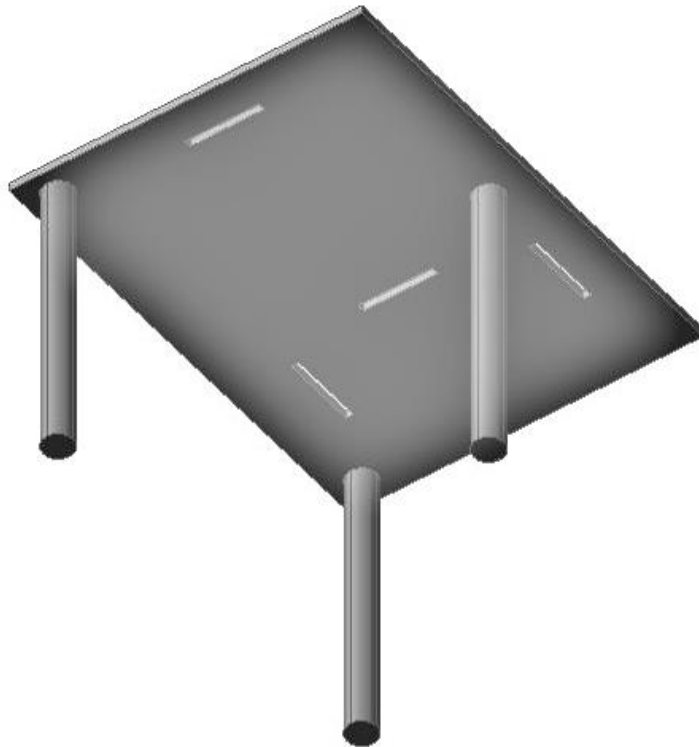
Le finestre consentono di alleggerire lo strumento e di facilitare l'assemblaggio dei componenti.



3° Fase: piegatura piastra

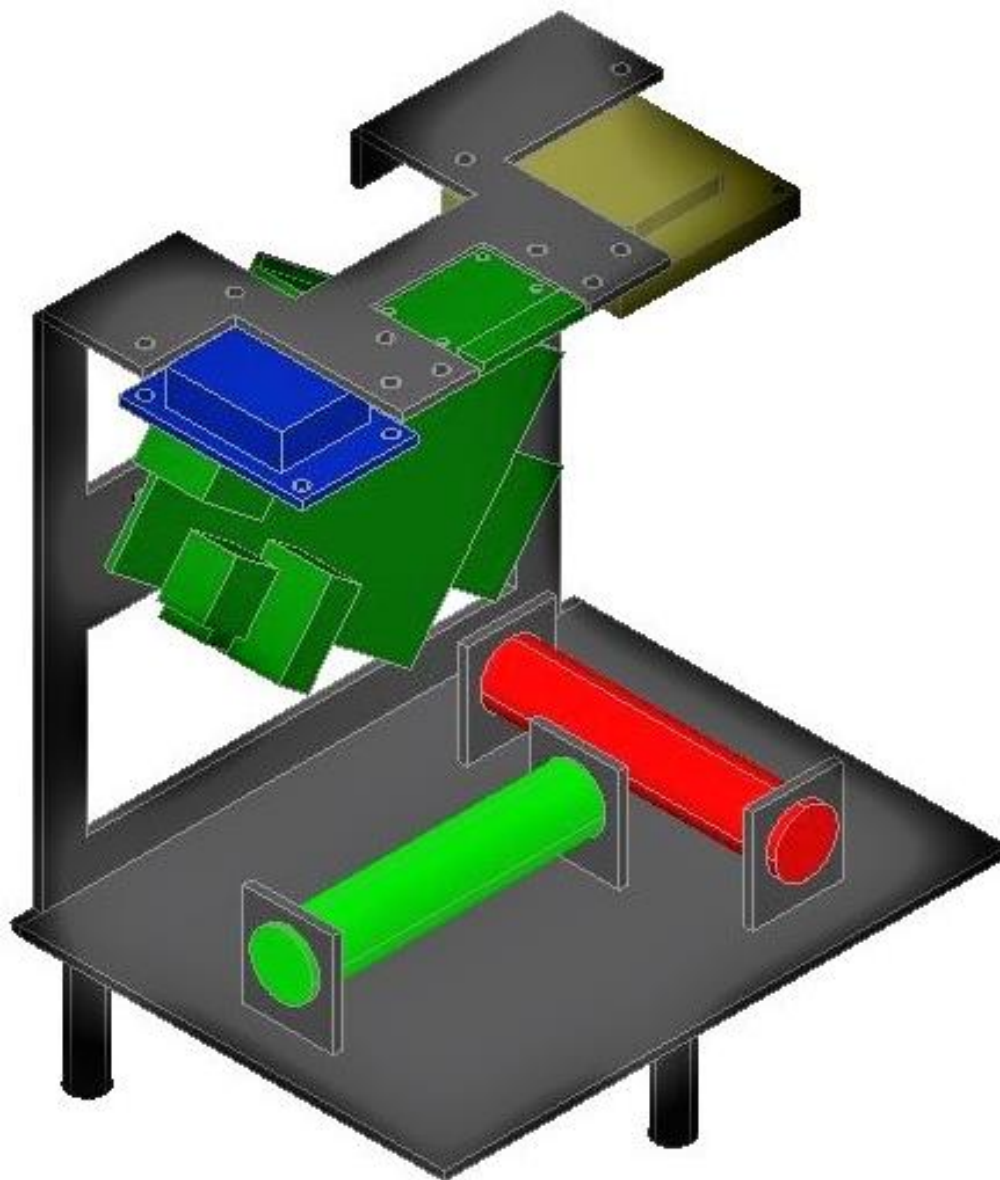


4° Fase: saldatura prigionieri

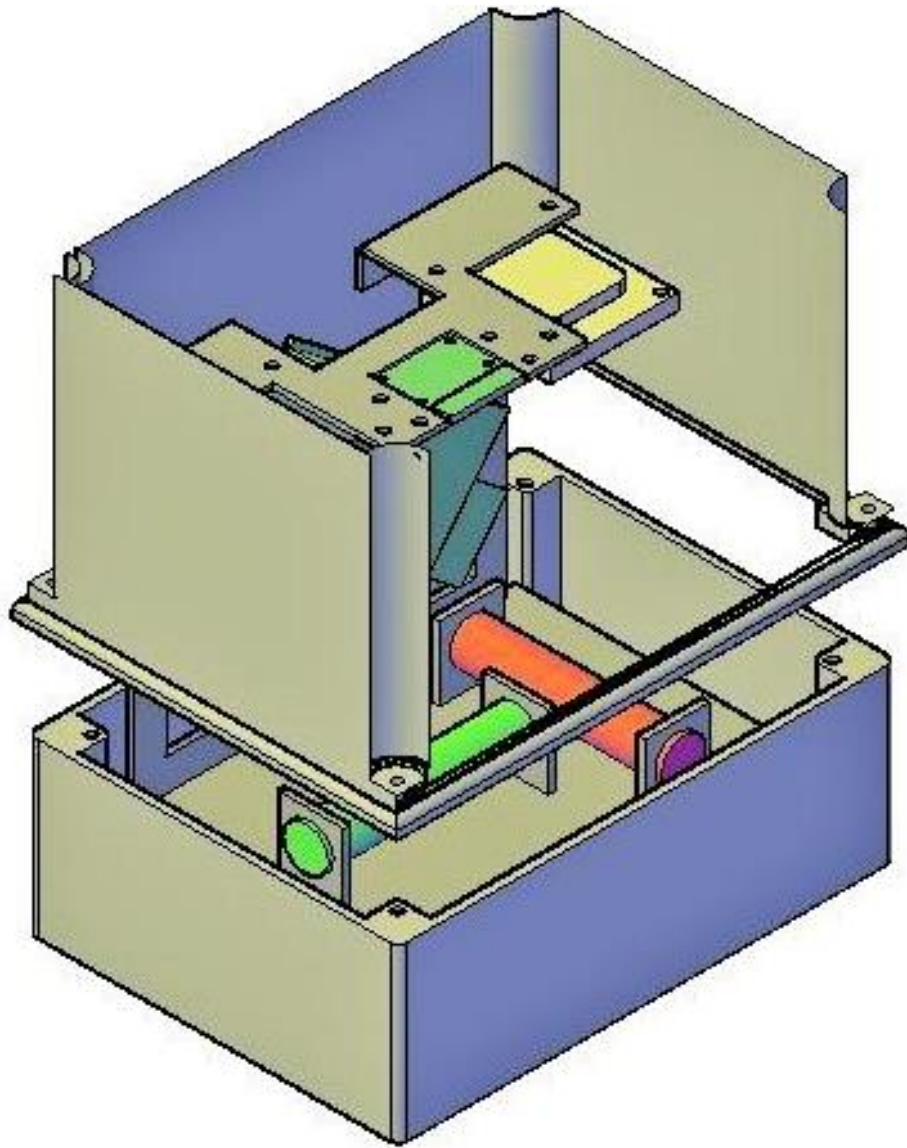


5° Fase: assemblaggio componenti

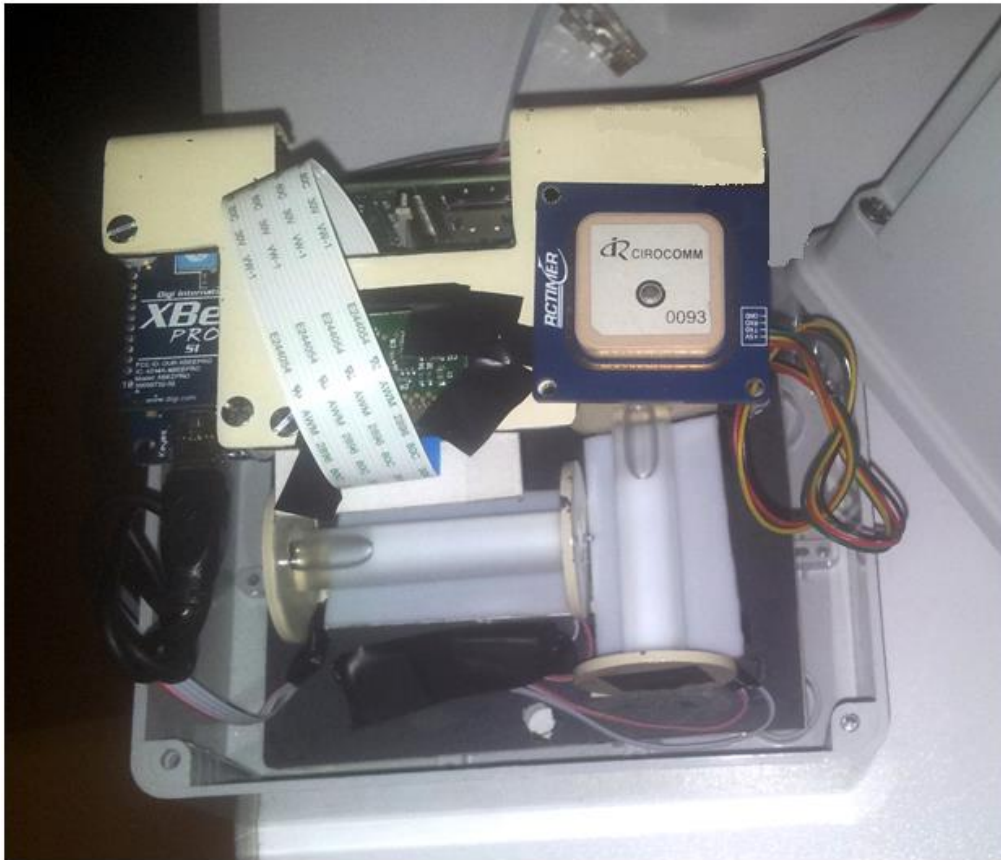
Gli elementi inseriti vengono bloccati tramite viti e relativi distanziatori per far sì che non vengano a contatto le parti metalliche.



6° Fase: inserimento nell'involucro







Deliverable Software

Sono stati messi a punto codici di calcolo per l'acquisizione e l'elaborazione dei dati. In particolare, sono stati realizzati i programmi di calcolo per:

- Calibrazione metrica delle fotocamere;
- Elaborazione delle immagini riprese dalle fotocamere e lettura clinometrica;
- Elaborazione dei rilievi GNSS.

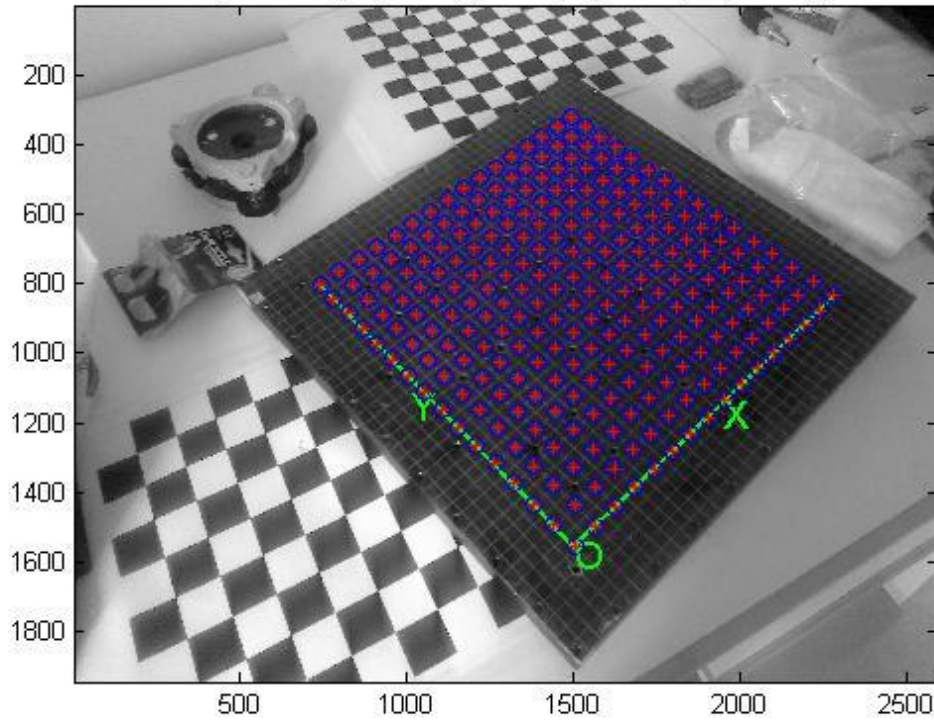
I programmi sono stati implementati nel processore.

2.4.2.1 CALIBRAZIONE METRICA DELLE FOTOCAMERE

Per la calibrazione metrica delle fotocamere è stato realizzato l'upgrade di una procedura già esistente sviluppata utilizzando Matlab[®]. La procedura può essere adoperata per qualsiasi fotocamera digitale.

Si è fatto uso, per le prove in laboratorio, di una piastra di calibrazione con accuratezza al decimo di mm. Come si può osservare dalla successiva figura, il software riconosce i punti di incrocio.

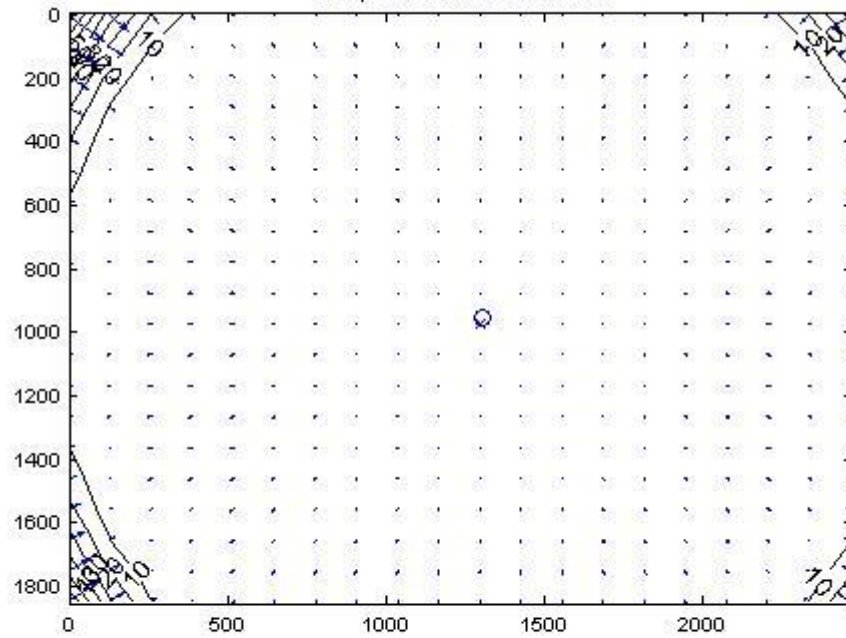
Image 1 - Image points (+) and reprojected grid points (o)



Questa operazione viene eseguita automaticamente, ma il posizionamento degli incroci è spesso poco preciso. Si è prevista, pertanto, la possibilità di intervento da parte dell'operatore che può correggere gli eventuali errori grossolani o eliminare dei falsi positivi individuati dalla procedura automatica.

I risultati ottenuti sono riportati nella figura seguente, dove sono riportati i parametri principali di una delle fotocamere testate.

Complete Distortion Model



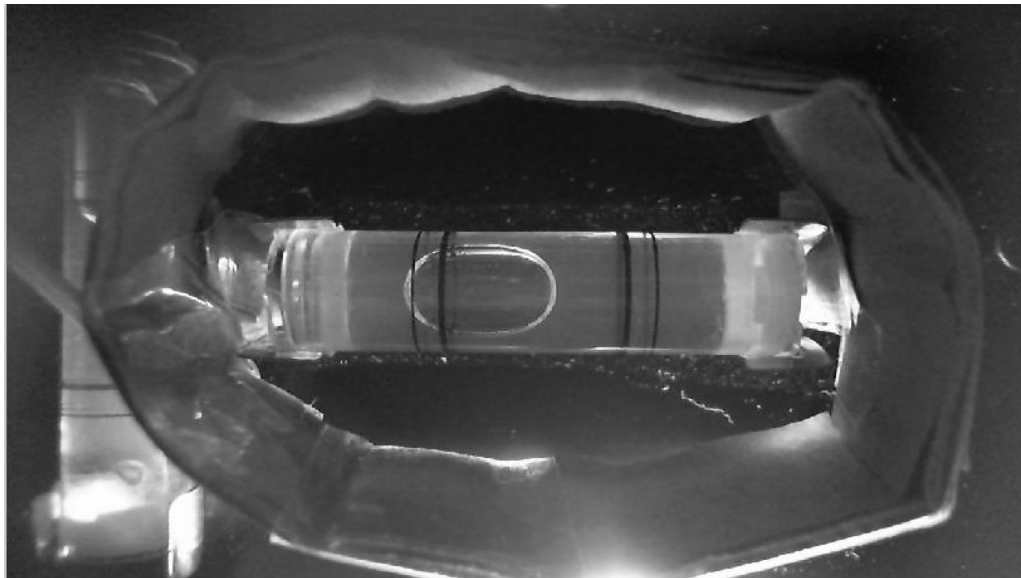
Pixel error	= [1.536, 1.628]	
Focal Length	= (2308.31, 2305.96)	+/- [6.581, 7.246]
Principal Point	= (1302.67, 954.307)	+/- [7.278, 6.465]
Skew	= 0.0016	+/- 0.0005117
Radial coefficients	= (0.06424, 0.003736, -0.722)	+/- [0.03211, 0.3828, 1.342]
Tangential coefficients	= (0.00079, -0.0001151)	+/- [0.001178, 0.001335]

2.4.2.2 ELABORAZIONE DELLE IMMAGINI RIPRESE DALLE FOTOCAMERE E LETTURA CLINOMETRICA

Inizialmente è stato scritto il codice in linguaggio Matlab[®], in modo da testare facilmente l'algoritmo e visualizzare i risultati.

In funzione della posizione reciproca delle livelle e della fotocamera, si ritaglia una porzione d'immagine per ognuna delle fiale, limitando l'analisi ad un'area ristretta del fotogramma, e cioè ad un intorno della livella della quale si vuole fare la lettura.

In una prima fase, si individuano grossolanamente i margini della bolla, considerando i valori dei pixel in scala di grigio. Si individuano i punti di bordo candidati, adoperando un valore di soglia. Non sono stati adoperati operatori "edge detectors" per limitare l'impegno computazionale.



Si esegue un affinamento sub-pixel della posizione dei punti trovati. Successivamente, interpolando questi punti con un'ovale, si ottiene una ricostruzione del bordo stesso.

Applicando il metodo dei minimi quadrati, si individua la curva interpolante che dev'essere una *sezione spirica* (sezione torica ottenuta con un piano parallelo

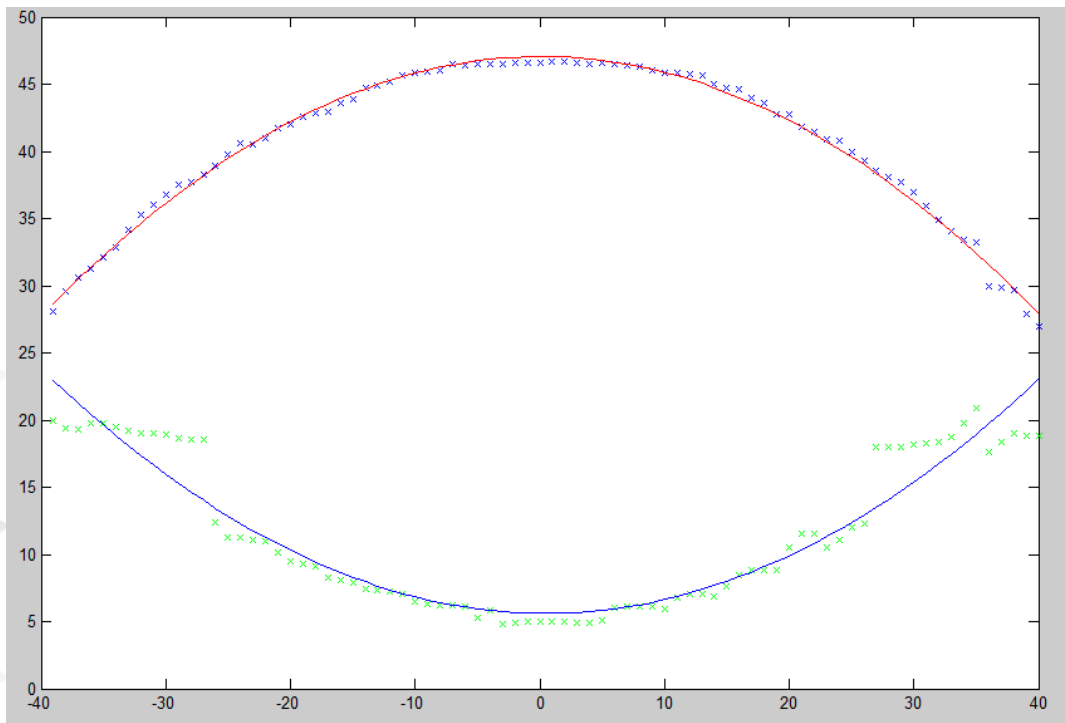
all'asse rotazionale). Nel nostro caso, la curva è del quarto ordine e la sua equazione è la seguente:

$$(r^2 - a^2 + c^2 + x^2 + y^2)^2 = 4r^2(x^2 + c^2)$$

in cui, avendo assunto il sistema di riferimento con origine nel centro del toro, y è l'asse rotazionale del toro, c è la z del piano di sezione, x l'asse che completa la terna di riferimento, r è il raggio dell'asse del toro, a il raggio del cerchio che genera il toro. Le incognite sono dovute alle incertezze di lavorazione (r ed a) ed alla variabilità della temperatura (c).

La soluzione ai minimi quadrati si ottiene considerando i valori di primo tentativo ottenuti.

Applicando tale metodo all'immagine precedente, si ottengono i risultati rappresentati nella figura successiva.



Le crocette rappresentano i punti corrispondenti al bordo della bolla, trovati sull'immagine analizzata. Rispettivamente le crocette verdi sono relative al bordo sinistro, quelle blu al bordo destro.

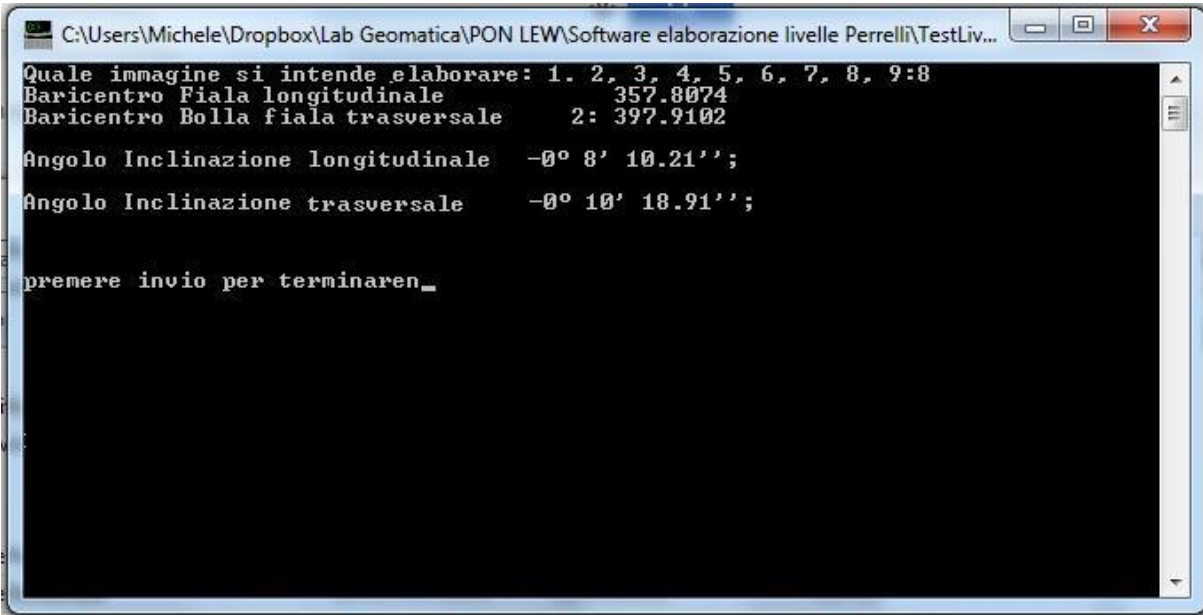
Le linee rappresentano l'ovale interpolante, che permette così la lettura finale.

Infatti i punti estremi dell'ovale rappresentano la lettura desiderata.

Si può osservare l'effetto dovuto alla presenza della tacca nella parte sinistra della foto. Il risultato finale è debolmente influenzato dalla tacca stessa e la differenza con l'altro estremo è inferiore all'approssimazione di lettura previsto per lo strumento.

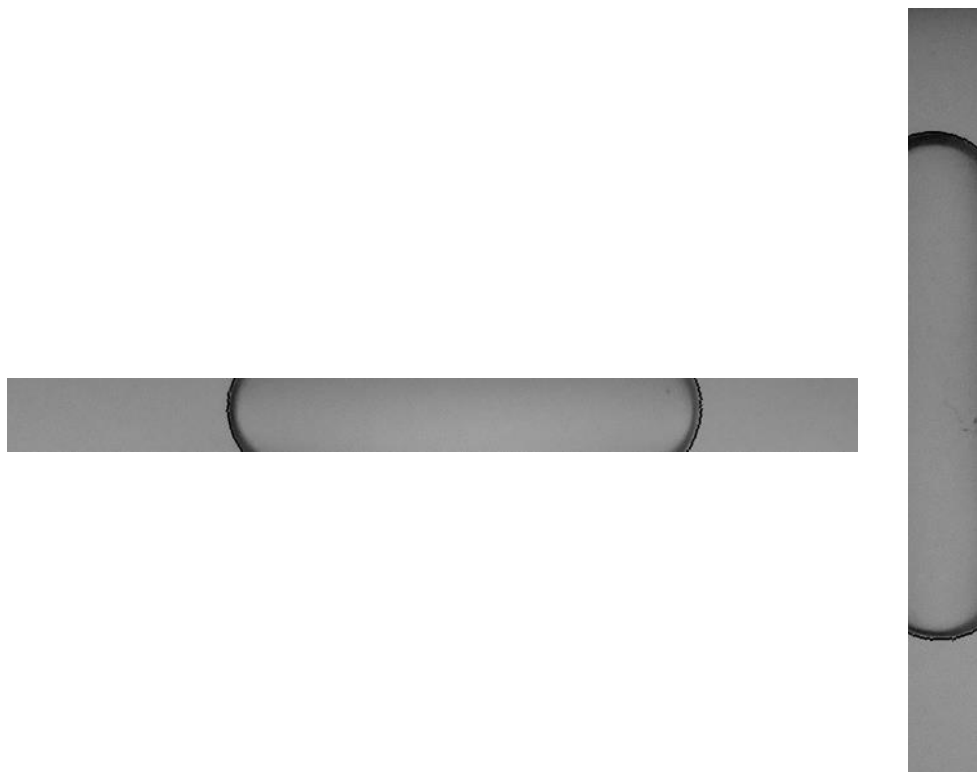
L'algoritmo scritto in Matlab[®] è stato in linguaggio C, in modo da poter essere eseguito da un microcontrollore.

Nella figura seguente osserviamo i risultati di una lettura, con le coordinate pixel del baricentro della bolla e l'inclinazione misurata dalle livelle in direzione longitudinale e trasversale espresse in gradi sessagesimali.



```
C:\Users\Michele\Dropbox\Lab Geomatica\PON LEW\Software elaborazione livelle Perrelli\TestLiv...
Quale immagine si intende elaborare: 1. 2. 3. 4. 5. 6. 7. 8. 9:8
Baricentro Fiala longitudinale 357.8074
Baricentro Bolla fiala trasversale 2: 397.9102
Angolo Inclinazione longitudinale -0° 8' 10.21'';
Angolo Inclinazione trasversale -0° 10' 18.91'';
premere invio per terminaren_
```

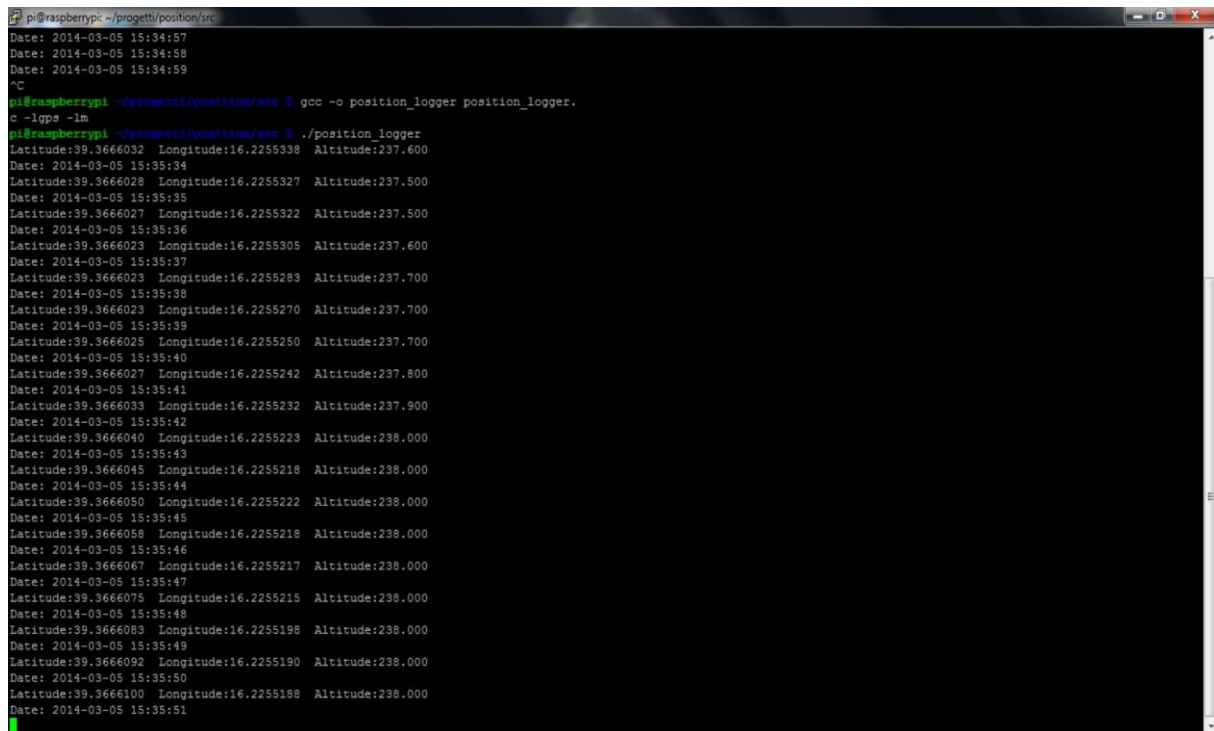

Lo strumento ha anche la possibilità di visualizzare la porzione d'immagine analizzata per l'elaborazione.



2.4.2.3 ELABORAZIONE DEI RILIEVI GNSS

Il codice messo a punto consente di iniziare e terminare l'acquisizione della posizione da parte del ricevitore GNSS installato. Si possono settare la durata dell'acquisizione, la frequenza di campionamento. In alternativa si può interrompere l'acquisizione solo una volta stabilizzato il valore delle coordinate ottenute.

Nella figura seguente è riportata una schermata con i valori delle coordinate geografiche acquisiti dal ricevitore ad intervalli di un secondo.



```
pi@raspberrypi: ~/progetti/position/src
Date: 2014-03-05 15:34:57
Date: 2014-03-05 15:34:58
Date: 2014-03-05 15:34:59
^C
pi@raspberrypii ~/progetti/position/src $ gcc -o position_logger position_logger.c -lpps -lm
pi@raspberrypii ~/progetti/position/src $ ./position_logger
Latitude:39.3666032 Longitude:16.2255338 Altitude:237.600
Date: 2014-03-05 15:35:34
Latitude:39.3666028 Longitude:16.2255327 Altitude:237.500
Date: 2014-03-05 15:35:35
Latitude:39.3666027 Longitude:16.2255322 Altitude:237.500
Date: 2014-03-05 15:35:36
Latitude:39.3666023 Longitude:16.2255305 Altitude:237.600
Date: 2014-03-05 15:35:37
Latitude:39.3666023 Longitude:16.2255283 Altitude:237.700
Date: 2014-03-05 15:35:38
Latitude:39.3666023 Longitude:16.2255270 Altitude:237.700
Date: 2014-03-05 15:35:39
Latitude:39.3666025 Longitude:16.2255250 Altitude:237.700
Date: 2014-03-05 15:35:40
Latitude:39.3666027 Longitude:16.2255242 Altitude:237.800
Date: 2014-03-05 15:35:41
Latitude:39.3666033 Longitude:16.2255232 Altitude:237.900
Date: 2014-03-05 15:35:42
Latitude:39.3666040 Longitude:16.2255223 Altitude:238.000
Date: 2014-03-05 15:35:43
Latitude:39.3666045 Longitude:16.2255218 Altitude:238.000
Date: 2014-03-05 15:35:44
Latitude:39.3666050 Longitude:16.2255222 Altitude:238.000
Date: 2014-03-05 15:35:45
Latitude:39.3666058 Longitude:16.2255218 Altitude:238.000
Date: 2014-03-05 15:35:46
Latitude:39.3666067 Longitude:16.2255217 Altitude:238.000
Date: 2014-03-05 15:35:47
Latitude:39.3666075 Longitude:16.2255215 Altitude:238.000
Date: 2014-03-05 15:35:48
Latitude:39.3666083 Longitude:16.2255198 Altitude:238.000
Date: 2014-03-05 15:35:49
Latitude:39.3666092 Longitude:16.2255190 Altitude:238.000
Date: 2014-03-05 15:35:50
Latitude:39.3666100 Longitude:16.2255188 Altitude:238.000
Date: 2014-03-05 15:35:51
```

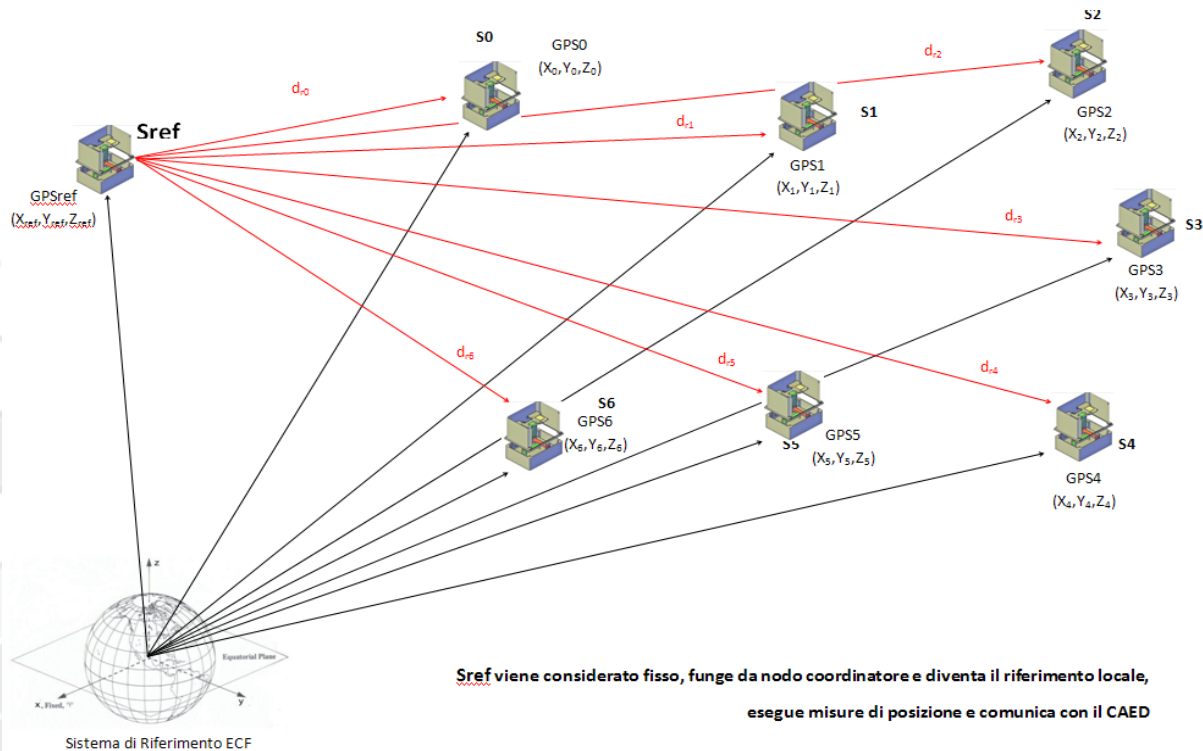
Per gli scopi di monitoraggio è preferibile adoperare le coordinate spaziali ortogonali. Un esempio di acquisizione ad intervalli di un secondo è riportata di seguito:

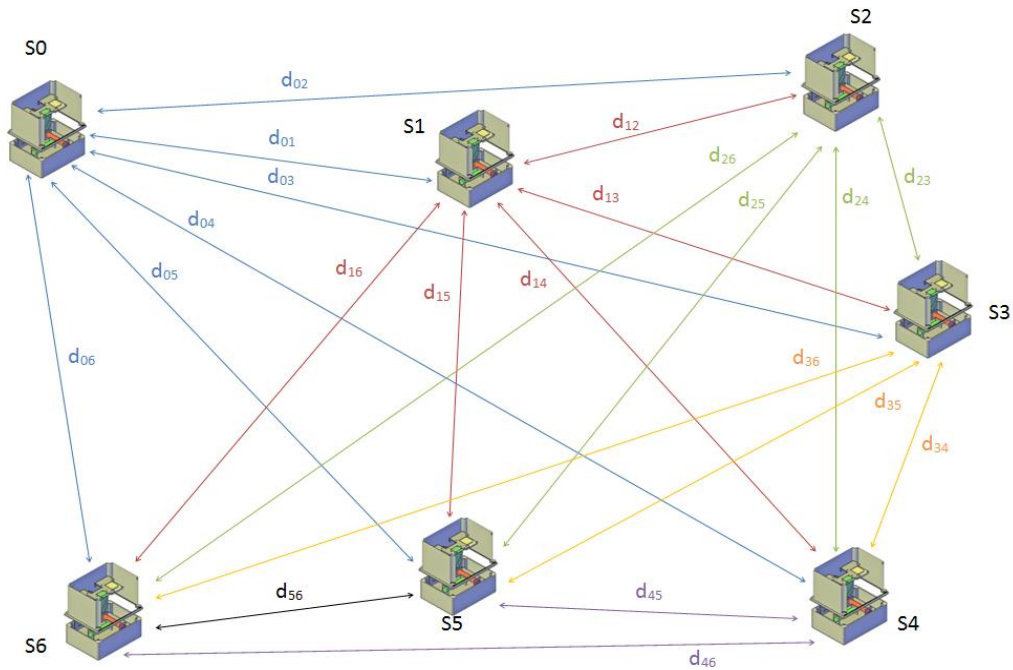
```

Date: 2014-03-07 16:35:59 X:4741161.958339 Y:1379711.126417 Z:4023925.290908
Date: 2014-03-07 16:36:00 X:4741161.958339 Y:1379711.126417 Z:4023925.290908
Date: 2014-03-07 16:36:01 X:4741161.962433 Y:1379711.112349 Z:4023925.290908
Date: 2014-03-07 16:36:02 X:4741161.955467 Y:1379711.095961 Z:4023925.304641
Date: 2014-03-07 16:36:03 X:4741161.933699 Y:1379711.044748 Z:4023925.347558
Date: 2014-03-07 16:36:04 X:4741161.922750 Y:1379710.996684 Z:4023925.376742
Date: 2014-03-07 16:36:05 X:4741161.920119 Y:1379710.965401 Z:4023925.390475
Date: 2014-03-07 16:36:06 X:4741161.976435 Y:1379710.922550 Z:4023925.496820
Date: 2014-03-07 16:36:07 X:4741161.965486 Y:1379710.874485 Z:4023925.526004
Date: 2014-03-07 16:38:07 X:4741161.981912 Y:1379711.148535 Z:4023929.670244
Date: 2014-03-07 16:38:08 X:4741161.993407 Y:1379711.151880 Z:4023929.655652
Date: 2014-03-07 16:38:09 X:4741162.086099 Y:1379711.193215 Z:4023929.690755
Date: 2014-03-07 16:38:10 X:4741162.175678 Y:1379711.204922 Z:4023929.739591
Date: 2014-03-07 16:38:11 X:4741162.238414 Y:1379711.223179 Z:4023929.817611
Date: 2014-03-07 16:38:12 X:4741162.249909 Y:1379711.226524 Z:4023929.803019

```

L'utilizzo del ricevitore satellitare è volto, piuttosto che al posizionamento, alla determinazione delle variazioni morfologiche della superficie da monitorare. Si prevede, pertanto, la risoluzione di una rete ed il calcolo delle baselines.





Uno dei nodi è considerato fisso ed è fornito del modulo XBee che funge da coordinatore. In questo nodo convergono i dati acquisiti da tutti i ricevitori e viene eseguito il calcolo delle baselines.

Le coordinate dei sensori vengono memorizzate in una matrice che ha la seguente forma:

ID	X	Y	Z
Sref	X_{REF}	Y_{ref}	Z_{REF}
S0	X_0	Y_0	Z_0
S1	X_1	Y_1	Z_1
S2	X_2	Y_2	Z_2
S3	X_3	Y_3	Z_3
S4	X_4	Y_4	Z_4
S5	X_5	Y_5	Z_5
S6	X_6	Y_6	Z_6
S7	X_7	Y_7	Z_7
⋮	⋮	⋮	⋮
Sn	X_N	Y_N	Z_N

In giallo la Matrice "Posizioni" del codice C.

Tra i vari punti sono calcolate le distanze euclidee riportate nella tabella seguente:

$d_{ij} = \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2 + (Z_i - Z_j)^2}$	S0	S1	S2	S3	S4	S5	S6
Sref	d _{r0}	d _{r1}	d _{r2}	d _{r3}	d _{r4}	d _{r5}	d _{r6}
S0		d ₀₁	d ₀₂	d ₀₃	d ₀₄	d ₀₅	d ₀₆
S1			d ₁₂	d ₁₃	d ₁₄	d ₁₅	d ₁₆
S2				d ₂₃	d ₂₄	d ₂₅	d ₂₆
S3					d ₃₄	d ₃₅	d ₃₆
S4						d ₄₅	d ₄₆
S5							d ₅₆

Si riporta di seguito la parte di codice in linguaggio C per l'organizzazione ed il calcolo delle baselines.

ALGORITMO CALCOLO BASELINES in C:

```
int i,j,k,dX,dY,dZ; %i=0 corrisponde a Sref
int n; %numero sensori
int numDistanze= fattoriale(n)/(2* fattoriale(n-2)); % n!/(m!*(n-m)!) Dove m=2
volendo trovare la mutua distanza tra due punti della rete
float distanze[numDistanze];
k=0;
for (i=0; i<n-1; i++)
{
    for (j=i+1; j<n; j++)
    {
        dX=Posizioni[i][0]- Posizioni[j][0];
        dY=Posizioni[i][1]- Posizioni[j][1];
        dZ=Posizioni[i][2]- Posizioni[j][2];
        distanze[k]=sqrt((dX*dX)+ (dY*dY)+ (dZ*dZ));
    }
}
```

Dove la funzione fattoriale è

```
int fattoriale(int n)
{
    if (n<=1)
        return(1);
    else
        n=n*fattoriale(n-1);
    return(n);
}
```

